AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application:

LISTING OF CLAIMS:

1. (currently amended) A serial communication device bridging between [[a]] an *m* bit parallel bus and a serial bus, comprising:

a parallel bus interface circuit that receives as an input *m* bit wide data from the parallel bus and multiplexes the *m* bit wide data into sequentially generated *n* bit wide parallel data segments, with $n < m$;

~~(a)~~ a check bit producer that receives as an input the *n* bit wide parallel data segments and produces as an output a parallel arrangement of the *n* bit wide parallel data segments and a generated ~~which applies an~~ error correcting code ~~to parallel data transmitted through said parallel bus~~; and

~~(b)~~ a parallel-serial converter which converts said parallel arrangement of the *n* bit wide parallel data segments and the error correcting code ~~data output~~ from said check bit producer[[,]] into serial data.

2. (canceled)

3. (currently amended) A serial communication device bridging between a parallel bus and a serial bus, comprising:

~~(a)~~ a serial-parallel converter which converts serial data with an error correcting code transmitted through said serial bus[[,]] into parallel arrangement of the *n* bit wide parallel data segments and the error correcting code; ~~and~~

~~(b)~~ an error detector which checks ~~an~~ the error correcting code ~~applied to~~ within said parallel ~~serial~~ data; and ~~, and detects an error in said error correcting code~~

a parallel bus interface circuit that demultiplexes the *n* bit wide parallel data segments from the error detector into *m* bit wide parallel data on the parallel bus.

4. (original) The serial communication device as set forth in claim 3, wherein said error detector has a function of correcting said error when said error is detected by said error detector.

5. (original) The serial communication device as set forth in claim 4, wherein said error detector corrects said error when said error is a 1-bit error, and abandons an access when said error is a 2-bit error.

6. (currently amended) A serial communication device bridging between a parallel bus and a serial bus, comprising:

a parallel bus interface circuit that receives as an input *m* bit wide data from the parallel bus and multiplexes the *m* bit wide data into sequentially generated *n* bit wide parallel data segments, with $n < m$;

~~(a)~~ a check bit producer that receives as an input the *n* bit wide parallel data segments and produces as an output a parallel arrangement of the *n* bit wide parallel data segments and a generated ~~which applies~~ an error correcting code ~~to parallel data transmitted through said parallel bus~~;

~~(b)~~ a parallel-serial converter which converts said parallel arrangement of the *n* bit wide parallel data segments and the error correcting code ~~data output~~ from said check bit producer[[,]] into serial data;

~~(c)~~ a serial-parallel converter which converts serial data with the error correcting code transmitted through said serial bus[[,]] into parallel arrangement of the *n* bit wide parallel data segments and the error correcting code; and

~~(d)~~ an error detector which checks ~~an~~ the error correcting code ~~applied to~~ within said parallel ~~serial~~ data~~, and detects an error in said error correcting code~~;

wherein the parallel bus interface is also connected to receive as an input the parallel data segments from the error detector, the parallel bus interface demultiplexing the *n* bit wide parallel data segments from the error detector into *m* bit wide parallel data on the parallel bus.

7. (canceled)

8. (original) The serial communication device as set forth in claim 6, wherein said error detector has a function of correcting said error when said error is detected by said error detector.

9. (original) The serial communication device as set forth in claim 6, wherein said error detector corrects said error when said error is a 1-bit error, and abandons an access when said error is a 2-bit error.

10. (currently amended) A method of carrying out serial communication between a parallel bus and a serial bus, comprising the steps of:

multiplexing *m* bit wide parallel data sequentially into *n* bit wide parallel data segments, where *m* > *n*;

~~(a)~~ applying an error correcting code to each *n* bit wide parallel data segment ~~data transmitted through said parallel bus~~; and

~~(b)~~ converting said parallel data with the error correcting code into serial data.

11. (canceled)

12. (currently amended) A method of carrying out serial communication between a parallel bus and a serial bus, comprising the steps of:

~~(a)~~ converting serial data with an included error correcting code into parallel arrangement of the *n* bit wide parallel data segments and the error correcting code;

~~(b)~~ checking ~~an~~ the error correcting code applied to each said ~~serial~~ parallel data segment; ~~and~~

~~(c)~~ ~~detecting~~ checking for an error ~~in~~ based on said error correcting code; and

demultiplexing the *n* bit wide parallel data segments into *m* bit wide parallel data on the parallel bus, wherein $m > n$.

13. (currently amended) The method as set forth in claim 12, further comprising the step of ~~(d)~~ correcting said error detected in said error checking step ~~(c)~~.

14. (original) The method as set forth in claim 12, further comprising the steps of:

(d) correcting said error when said error is a 1-bit error; and

(e) abandoning an access when said error is a 2-bit error.

15. (currently amended) A method of carrying out serial communication between a parallel bus and a serial bus, comprising the steps of:

when transferring data from the parallel bus to the serial bus:

multiplexing *m* bit wide parallel data from the parallel bus into *n* bit wide data segments, where *m* > *n*;

~~(a)~~ applying an error correcting code to each parallel data segment ~~transmitted through said parallel bus~~; and

~~(b)~~ converting each said parallel data segment with the error code into serial data; and

when transferring data from the serial bus to the parallel bus:

~~(c)~~ converting serial data with included error codes transmitted through said serial bus[[,]]into parallel arrangement of the *n* bit wide parallel data segments and the error correcting code;

~~(d)~~ ~~checks~~ checking ~~an~~ the error correcting code applied to each said ~~serial~~ parallel data segment; ~~and~~

~~(e)~~ detecting an error in said error correcting code; and

demultiplexing the *n* bit wide parallel data segments into *m* bit wide parallel data on the parallel bus.

16. (canceled)

17. (currently amended) The method as set forth in claim 15, further comprising the step of ~~(f)~~ correcting said error detected in said error detecting step ~~(e)~~.

18. (original) The method as set forth in claim 15, further comprising the steps of:

(f) correcting said error when said error is a 1-bit error; and

(g) abandoning an access when said error is a 2-bit error.

19. (new) The device of claim 1, wherein $m = 32$ and $n = 8$.

20. (new) The device of claim 3, wherein $m = 32$ and $n = 8$.

21. (new) The device of claim 6, wherein $m = 32$ and $n = 8$.

22. (new) The method of claim 10, wherein $m = 32$ and $n = 8$.

23. (new) The method of claim 12, wherein $m = 32$ and $n = 8$.

24. (new) The method of claim 15, wherein $m = 32$ and $n = 8$.

25. (new) The device of claim 6, wherein data that is transferred from the parallel bus interface circuit to the check bit producer travels along a different path than does data that is transferred from the error detector to the parallel bus interface circuit.

26. (new) The method of claim 15, wherein the n bit wide data segments transferred while communicating from the parallel bus to the serial bus follow a different path than that used to transfer the n bit wide data segments while communicating from the serial bus to the parallel bus.